

EV316936733

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR LETTERS PATENT

Centralized Detector of Dynamic, Robust, Embedded-Signals

Inventor(s):

Theodore C. Tanner Jr.

Steven Swenson

Martin G. Puryear

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324.9256
F: 509.323.8979
www.leehayes.com



ATTORNEY's DOCKET NO. MS1-1348US

1 CENTRALIZED DETECTOR OF DYNAMIC, ROBUST, EMBEDDED-
2 SIGNALS

3 TECHNICAL FIELD

4
5 This invention generally relates to a technology for detection of one or
6 more embedded-signals.

7 BACKGROUND

8
9 As used herein, “intangible goods” is a generic label for electronically
10 stored or transmitted content. Examples of intangible goods include images, audio
11 clips, video, multimedia, software, metadata, and data. An intangible good may be
12 analog or digital. Depending upon the context, an intangible goods may also be
13 called a “digital signal,” “content signal,” “digital bitstream,” “media signal,”
14 “digital object,” “object,” and the like.

15
16 Intangible goods are often distributed to consumers over private and public
17 networks—such as Intranets and the Internet. In addition, these goods are
18 distributed to consumers via fixed computer-readable media, such as a compact
19 disc (CD-ROM), digital versatile disc (DVD), soft magnetic diskette, hard
20 magnetic disk (e.g., a preloaded hard drive), portable media players, and flash
21 memory cards. Furthermore, goods are distributed via communications streams
22 such as those originating from a client such as an instant messenger or another
23 audio/visual chat application.

24
25 Unfortunately, it is relatively easy for a person to pirate the content of
intangible goods at the expense and harm of the content owners—which include
the content author, publisher, developer, distributor, etc. The content-based

1 industries (e.g., entertainment, software, audio and/or video, film, etc.) that
2 produce and distribute content are plagued by lost revenues due to piracy.

3

4 **Embedded-Signals**

5 Embedding one or more signals in a carrier signal (e.g., intangible goods) is
6 one of the most promising techniques for protecting the content owner's rights of
7 intangible goods. This embedded-signal is commonly called a "watermark" and
8 the embedding process is commonly called "watermarking."

9 Generally, watermarking is a process of altering the intangible good such
10 that its perceptual characteristics are preserved. For example, a "watermark" is a
11 pattern of bits or signal stream inserted into a digital or analog good that may be
12 used for many purposes, such as identifying the content owners and/or the
13 protected rights.

14 A watermark embedder (i.e., encoder) is used to embed a watermark into
15 intangible goods. A watermark detector is used to detect the existence of the
16 watermark in the watermarked intangible goods and possibly identifying that
17 watermark.

18 Watermark detection is often performed in real-time even on small
19 electronic components. Such a "real-time" detector is also often called a
20 "dynamic detector." Generally, this means that the detector is attempting to detect
21 a watermark in intangible goods as the goods are being consumed (e.g., played,
22 presented, stored, and such). For example, if the intangible good is an audio
23 signal, the detector attempts detection while the audio signal is being played.

24 Such dynamic watermark detection is often a very expensive operation (in
25 terms of computing resources). If there are multiple input streams, then

1 conventionally there are multiple dynamic watermark detection modules running
2 (i.e. one per input stream). The expense in computing resources increases with
3 each watermark detection module invoked to operate on an input stream.

4 Those of ordinary skill in the art are familiar with conventional techniques
5 and technology associated with watermarks, watermark embedding, and
6 watermark detecting.

7 **SUMMARY**

9 Described herein is a technology for dynamic and robust detection of one or
10 more embedded-signals (e.g., watermark, copyright notice, encoded data, etc.) in
11 an input signal (e.g., multimedia stream, video stream, audio stream, data, radio,
12 etc.) within a multi-signal environment.

13 This summary itself is not intended to limit the scope of this patent.
14 Moreover, the title of this patent is not intended to limit the scope of this patent.
15 For a better understanding of the present invention, please see the following
16 detailed description and appending claims, taken in conjunction with the
17 accompanying drawings. The scope of the present invention is pointed out in the
18 appending claims.

19 **BRIEF DESCRIPTION OF THE DRAWINGS**

21 The same numbers are used throughout the drawings to reference like
22 elements and features.

23 Fig. 1 is a schematic illustration of a multi-signal environment within which
24 an implementation described herein may operate.

1 Fig. 2 is a block diagram showing a production and distribution system in
2 which a content producer/provider watermarks intangible goods and subsequently
3 distributes that watermarked intangible goods to a client. It also shows the
4 computer client capable of (wholly or partially) implementing at least one
5 embodiment described herein.

6 Fig. 3 is a block diagram illustrating an example of the operation of an
7 implementation described herein.

8 Fig. 4 is a flow diagram showing a methodological implementation
9 described herein.

10 Fig. 5 is an example of a computing operating environment capable of
11 (wholly or partially) implementing at least one embodiment described herein.

12 **DETAILED DESCRIPTION**

13 In the following description, for purposes of explanation, specific numbers,
14 materials and configurations are set forth in order to provide a thorough
15 understanding of the present invention. However, it will be apparent to one skilled
16 in the art that the present invention may be practiced without the specific
17 exemplary details. In other instances, well-known features are omitted or
18 simplified to clarify the description of the exemplary implementations of the
19 present invention, thereby better explaining the present invention. Furthermore, for
20 ease of understanding, certain method steps are delineated as separate steps;
21 however, these separately delineated steps should not be construed as necessarily
22 order dependent in their performance.

23 The following description sets forth one or more exemplary
24 implementations of a Centralized Detector of Dynamic, Robust, Embedded-
25

1 Signals that incorporate elements recited in the appended claims. These
2 implementations are described with specificity in order to meet statutory written
3 description, enablement, and best-mode requirements. However, the description
4 itself is not intended to limit the scope of this patent.

5 The inventors intend these exemplary implementations to be examples. The
6 inventors do not intend these exemplary implementations to limit the scope of the
7 claimed present invention. Rather, the inventors have contemplated that the
8 claimed present invention might also be embodied and implemented in other ways,
9 in conjunction with other present or future technologies.

10 An example of an embodiment of a Centralized Detector of Dynamic,
11 Robust, Embedded-Signals may be referred to as an “exemplary central watermark
12 detector” or “exemplary CWD.”

13 **Incorporation by Reference**

14 The following co-pending patent application is incorporated by reference
15 herein: U.S. Patent Application Serial No. ___, entitled “Circumvention of
16 Dynamic, Robust, Embedded-Signal Detection” filed on ___, and assigned to the
17 Microsoft Corporation.
18

19 **Introduction**

20 The one or more exemplary implementations, described herein, of the
21 present claimed invention may be implemented (in whole or in part) by a central
22 watermark detector 252 as part of a computing device 226 (of Fig. 2) and/or as
23 part of a computing environment like that shown in Fig. 5.
24

1 The central watermark detector 252 may be implemented with software,
2 hardware, or a combination thereof. More specifically, this detector may be part
3 of an operating system 250 (as shown in Fig. 2).

4

Multi-Signal Environment

5 Fig. 1 illustrates a typical multi-signal environment 100, where a computing
6 device 110 receives input from multiple sources—each source has one or more
7 signals that potentially includes watermarked intangible goods. A multi-signal
8 environment can result from one source that is capable of generating multiple
9 input signals. An input signal includes signals that may be unmodified or
10 modified within the system receiving them.

11 Examples of sources that may send one or more input signals include the
12 following illustrated in Fig. 1:

- 14 • a video camera 112;
- 15 • a CD or DVD player 114;
- 16 • other video source 116 (e.g., tape, hard drive, video streaming, etc.);
- 17 • another computer system 118 located on a network 120 (such as the
18 Internet);
- 19 • a midi keyboard or device 122;
- 20 • digital or analog audio input 124 from any of many different sources
21 and formats (e.g., mp3 files, audio channels from video games, etc.);
- 22 • removable or non-removable data storage devices 128 (e.g., floppy
23 disks, hard drive, flash memory, USB devices, etc.)

1 Many of the devices can produce multiple input channels. For example,
2 line 126 is labeled “x 256” to indicate that it may actually be 256 different input
3 audio channels.

4 To protect the incoming intangible goods (e.g., licensed music or video),
5 watermark detection must be performed on each incoming signal. In the case of
6 hundreds of input channels, that means dynamic watermark detection must be
7 performed on each of the hundreds of channels while the signals are being
8 received.

9 Traditionally, each separate possible input has its own watermark detector.
10 Often this was accomplished by using a separate watermark detector program
11 module (typically software, but may be hardware as well) for each separate input.
12 This conventional approach requires a separate module for each input port.

13 For example, the CD player 114 may have its own watermark detection
14 program module loaded into memory by the operating system of the computing
15 device 110. Also, the multimedia player (for playing streaming video from the
16 Internet) may have its own separate watermark detection program module. And so
17 forth for each possible separate input.

18 As illustrated by line 126 of Fig. 1, some conventional video games may
19 have as many as 256 channels of audio input. The conventional approach requires
20 256 separate instances of the watermark detector running on the computing device
21 110. Each one tests the signal from an assigned input channel.

22 Although they are illustrated as separate hardware devices, input signals
23 may be generated by software program modules (e.g., a threads) running on the
24 computing device 110.

1 Thus, the conventional approach to watermark detection in a multi-signal
2 environment is inefficient because watermark detection is performed separately
3 for each input. Thus, there is much duplication of effort. In an alternative
4 conventional approach there are detectors on only some of the inputs. In that
5 instance, a pirate may defeat the conventional approach by rerouting input around
6 its dedicated detector.

7 The computing device 110 may also have one or more output paths 130.
8 These output paths may be used, for example, to communicate with a network
9 device, burning a Redbook CD, playing the S/PDIF port to an external digital
10 receiver (recorder), etc.

11 **Production and Distribution System Employing Embedded-Signals**

12 Fig. 2 shows an example of a content production and distribution system
13 220 having a content producer/provider 222 that produces original content (e.g.,
14 original audio and/or video) and distributes the content over a network 224 to the
15 computing device 226 or via computer-readable media 225, such as a CD-ROM.
16

17 The content producer/provider 222 has a content storage 230 to store
18 intangible goods (e.g., multimedia streams) of original content. The content
19 producer 222 has a watermark encoding system 232 to embed the intangible goods
20 with a watermark. That watermark may uniquely identify the content with the
21 content producer/provider 222. The watermark encoding system 232 may be
22 implemented as a standalone process or incorporated into other applications or an
23 operating system.

1 The watermark encoding system 232 applies the watermark to intangible
2 goods from the content storage 230. The watermark may, for example, identify
3 the content producer 222 by providing a signature that is embedded in the signal.

4 The content producer/provider 222 has a distribution server 234 that
5 streams the watermarked intangible goods over the network 224 (e.g., the
6 Internet). Alternatively, it stores the watermarked intangible goods onto computer-
7 readable media 225 (e.g., floppy disk or CD-ROM).

8 The computing device 226 is equipped with a processor 240, a memory
9 242, one or more input devices 246, and one or more output devices 244 (e.g.,
10 speakers, monitor, digital media writer, etc.). The one or more input devices is
11 designed to receive signals containing intangible goods from one or more sources,
12 such as sources illustrated in Fig. 1.

13 The memory 242 stores an operating system 250 (such as a Microsoft®
14 Windows XP® operating system), which executes on the processor. The
15 computing device 226 may be embodied in a many different ways, including a
16 computer, a handheld entertainment device, a set-top box, a television, an audio
17 appliance, and so forth.

18 Typically, multiple program modules are running under the operating
19 system 250. One module may be a central watermark detector (CWD) 252 which
20 is an implementation of the exemplary CWD described herein. With the CWD,
21 this computing device 226 only needs one watermark detector for its multiple
22 inputs coming through the one or more input devices 246.

23 Typically, the CWD 252 is a software module and it is typically
24 incorporated into the operating system. Alternatively, the CWD 252 may be
25 implemented in hardware which is called by the operating system 250.

1 Another module may be a intangible goods consumer 254, which is
2 designed to receive and consume the incoming intangible goods. Of course, the
3 consumer 254 does not need to be a part of the operating system. The consumer
4 254 may be, for example, in the case of multimedia, a multimedia player to
5 facilitate play of multimedia content through the output device(s) 244 (e.g., sound
6 card, speakers, storage unit, etc.). It may be, for example, a third-party driver for a
7 external device. If the watermark is present, the computing device can detect its
8 presence and identify its associated information.

9 Alternatively, block 254 could be a digital transceiver that conveys an
10 omnibus mixed signal to a receiver external to the client computer.

11 The operating system 250 and/or processor 240 may be configured to
12 enforce certain rules imposed by the content producer/provider (or copyright
13 owner). For instance, the operating system and/or processor may be configured to
14 reject fake or copied content that does not possess a valid watermark. In another
15 example, the system could play unverified content with a reduced level of fidelity.
16 In still another example, unverified content is played through analog outputs only.

17 The CWD 252 is a dynamic detector. Typically, when a dynamic detector
18 does not detect an embedded-signal, then it assumes that there is no embedded-
19 signal (e.g., watermark) in the subject intangible goods (e.g., carrier signal).

20 With a dynamic detector, the signal is being consumed (e.g., played, stored,
21 presented, etc.) while detection is being performed. If a watermark is detected, the
22 enforcement modules on the computing device 226 may impair (e.g., halt or mute)
23 the consumption of the incoming signal.

1 **Exemplary Central Watermark Detector (CWD)**

2 As illustrated in Fig. 3, the exemplary CWD logically views the input
3 signal pipeline as a tree, with multiple sources flowing downward from the top,
4 mixing as they flow to the bottom, where a single omnibus signal stream exists.
5 Generally speaking, an omnibus signal stream contains two or more incoming
6 signals mixed together. The exemplary CWD runs on the root of this tree.

7 Fig. 3 illustrates an exemplary multi-signal situation of the operation of the
8 exemplary CWD. All of the incoming signals are mixed together. As shown in
9 Fig. 3 at 320, game audio signal 310 and DVD audio signal 312 are mixed
10 together. Similarly, audio file signal 314 (e.g., a MP3-formatted file) and a midi
11 synthesis signal 316 are mixed together at 322. At 324, this mixed signal is added
12 to another audio signal 318, such as WMA file being played over a network.

13 Alternatively, Fig. 3 may be viewed as multiple video signals being mixed
14 together. In that view, a game video signal and DVD main video signal are mixed
15 together at 320 of Fig. 3. At 322, video subtitle signal and DVD menu signal are
16 mixed together. At 324, this mixed signal is added to one or more other signals,
17 such as closed caption data signal and any Picture-In-Picture video signals.

18 Each of these mixer points 320, 322, and 324 represents a “branch” in this
19 tree-like structure.

20 As shown in Fig. 3, a CWD 310 receives the omnibus mixed signal as
21 output from mixer 324. The CWD performs watermark detection on the omnibus
22 mixed signal. Thus, all of the incoming signals are mixed together before testing
23 by the CWD 310.

If no watermark is detected in the omnibus mixed signal, then that means that none of the original unmixed input signals (e.g., 310-318) have a watermark embedded therein. If it does indeed detect a watermark, then that means that at least one of the original unmixed input signals (e.g., 310-318) includes a watermark.

Of course, this assumes that mixing the signals does not destroy the watermarks embedded therein. Watermarks are typically designed to remain intact after purposeful attacks by pirates. Therefore, it is reasonable to assume that these watermarks are sufficiently robust to withstand mixing.

Amongst the multiple input signals, there may be multiple *watermarked* streams being played concurrently. So, the detector may be actively tracking more than one watermark at any time. Consequently, it may be in different parts of the mixing tree concurrently.

Searching for the Watermarked Signal

If the CWD 310 detects a watermark, it knows that at least one of the incoming signals has a watermark. As shown in Fig. 3, the mixing of the incoming signals may be viewed as a “tree.” As such, the CWD 310 may perform a breadth- or depth-search of the tree to locate the one incoming signal with the watermark. Those of ordinary skill in the art are familiar with such tree search approaches.

The CWD 310 performs a “tree search” to find the “branch” (of mixed inputs) that has the mark in it. In doing the tree search, the CWD 310 ultimately reaches the “leaf” (or single input) that has the watermark therein.

Once it locates the signal with the embedded signal, it notifies the operating system or an “enforcer.” Alternatively, the CWD 310 may be empowered to take action. Such action may include user-notification (e.g., audio and/or textual dialog box), transmission of a notification signal (to copyright clearinghouse, perhaps), and/or signal impairment. Other possible actions include impairing the master signal immediately while finding the correct “leaf” signal. As certain signals (or submixes) are ruled out, those may be re-enabled. Impairment may include, for example, muting an audio signal or severely degrading a multimedia signal.

Other Implementation Notes

It may take several seconds (10-15) upon initial reception to detect an incoming watermark. But since the exemplary CWD is processing the mixed signals (rather than each one separately and concurrently), it may process them all very quickly. Therefore, the CWD 310 is configured to determine whether an incoming signal includes a watermark or not within 30 seconds of the signal consumption begins.

The specific watermark detection engine utilized by the (CWD) is an implementation design choice. In fact, the exemplary CWD may utilize multiple different watermark detection engines to concurrently detect a watermark using the multiple ways in which a watermark may be embedded into a signal.

Watermark Circumvention Countermeasures

To avoid a “starvation” type watermark circumvention attack, the CWD may run at a higher priority than the media (e.g., audio or video) mixing. Then the CWD will never effectively get starved since before the system gets to that point,

1 the media signal would stutter and stop playing. Effectively, the CWD would be a
2 higher priority than the resultant media that is actually playing.

3 To avoid an “input-location and –interference” type watermark
4 circumvention attack, the CWD may move around in memory periodically. This
5 will make it difficult to get a fix on the signal’s input location.

6 To avoid a “circumvention by decoy” type attack, there may be multiple
7 concurrent CWDs running in different parts of the tree (and always one at the
8 root).

9 **Methodological Implementation of the Exemplary CWD**

10 Fig. 4 shows a methodological implementation of the exemplary CWD
11 (CWD) performed by the central watermark detector 252 (or some portion
12 thereof). This methodological implementation may be performed in software,
13 hardware, or a combination thereof.

14 At 410 of Fig. 4, the exemplary CWD receives multiple input signals—
15 each containing intangible goods that potentially includes an embedded-signal
16 (e.g., a watermark).

17 At 412, the exemplary CWD mixes together the signals in a tree-like
18 structure. As shown in Fig. 3, this tree-like structure is configured so that small
19 groups (e.g., 2-4) of signals are initially mixed and then those “branches” are
20 progressively mixed into small groups (e.g., 2-4) of branches until all of the
21 signals are mixed together into an omnibus mixed signal.

22 At 414, it performs detection on the omnibus mixed signal for an embedded
23 signal. If a watermark is not detected, then the process ends for now. Since the
24 inputs are dynamically changing and a watermark may be inserted anywhere
25

1 within a signal, this process is likely to effectively be endlessly repeated as long as
2 there are input signals. If a watermark is actually detected, then the process moves
3 to block 416.

4 Alternatively (or in addition to) moving to 416, the exemplary CWD may
5 impair the omnibus signal. As it eliminates “leafs” as potential suspects, it
6 releases them from impairment. Therefore, it eventually narrows the number of
7 streams under enforcement down to only the individual watermarked stream over
8 time as the process locates the targeted signal.

9 At 416, the exemplary CWD performs a progressive walk-up the tree-like
10 structure using a tree-search (e.g., breadth- or depth-search). Consequently, it
11 progressively “walks” its way through the tree-like structure in which the signals
12 are mixed. At each branch, it tests for a watermark. If it does not find one, it
13 moves on to another branch. If it finds one, then it moves up into that branch. It
14 continues that until it locates at least one signal with a watermark detected therein.
15 It may continue to test other branches to see if there is more than one signal
16 containing a watermark.

17 At 418, it indicates which signal(s) are located by the progressive walk-up
18 the tree-like structure using a tree-search. This indication may be to the operating
19 system or a specially designated enforcer program module.

20 At 420, the exemplary CWD may perform notification and/or enforcement.
21 Notification may include user-notification (e.g., audio and/or textual dialog box)
22 or transmission of a notification signal (to copyright clearinghouse, perhaps).
23 Enforcement may include signal impairment, such as muting an audio signal or
24 severely degrading a multimedia signal.

Exemplary Computing System and Environment

Fig. 5 illustrates an example of a suitable computing environment 500 within which an exemplary CWD, as described herein, may be implemented (either fully or partially). The computing environment 500 may be utilized in the computer and network architectures described herein.

The exemplary computing environment 500 is only one example of a computing environment and is not intended to suggest any limitation as to the scope of use or functionality of the computer and network architectures. Neither should the computing environment 500 be interpreted as having any dependency or requirement relating to any one or combination of components illustrated in the exemplary computing environment 500.

The exemplary CWD may be implemented with numerous other general purpose or special purpose computing system environments or configurations. Examples of well known computing systems, environments, and/or configurations that may be suitable for use include, but are not limited to, personal computers, server computers, thin clients, thick clients, hand-held or laptop devices, multiprocessor systems, microprocessor-based systems, set top boxes, programmable consumer electronics, network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, and the like.

The exemplary CWD may be described in the general context of computer-executable instructions, such as program modules, being executed by a computer. Generally, program modules include routines, programs, objects, components, data structures, etc. that performs particular tasks or implement particular abstract

1 data types. The exemplary CWD may also be practiced in distributed computing
2 environments where tasks are performed by remote processing devices that are
3 linked through a communications network. In a distributed computing
4 environment, program modules may be located in both local and remote computer
5 storage media including memory storage devices.

6 The computing environment 500 includes a general-purpose computing
7 device in the form of a computer 502. The components of computer 502 may
8 include, by are not limited to, one or more processors or processing units 504, a
9 system memory 506, and a system bus 508 that couples various system
10 components including the processor 504 to the system memory 506.

11 The system bus 508 represents one or more of any of several types of bus
12 structures, including a memory bus or memory controller, a peripheral bus, an
13 accelerated graphics port, and a processor or local bus using any of a variety of bus
14 architectures. By way of example, such architectures can include a CardBus,
15 Personal Computer Memory Card International Association (PCMCIA),
16 Accelerated Graphics Port (AGP), Small Computer System Interface (SCSI),
17 Universal Serial Bus (USB), IEEE 1394, a Video Electronics Standards
18 Association (VESA) local bus, 3GIO (from INTEL™), NuBus (from Apple™)
19 and a Peripheral Component Interconnects (PCI) bus also known as a Mezzanine
20 bus.

21 Computer 502 typically includes a variety of computer readable media.
22 Such media may be any available media that is accessible by computer 502 and
23 includes both volatile and non-volatile media, removable and non-removable
24 media.

1 The system memory 506 includes computer readable media in the form of
2 volatile memory, such as random access memory (RAM) 510, and/or non-volatile
3 memory, such as read only memory (ROM) 512. A basic input/output system
4 (BIOS) 514, containing the basic routines that help to transfer information
5 between elements within computer 502, such as during start-up, is stored in ROM
6 512. RAM 510 typically contains data and/or program modules that are
7 immediately accessible to and/or presently operated on by the processing unit 504.

8 Computer 502 may also include other removable/non-removable,
9 volatile/non-volatile computer storage media. By way of example, Fig. 5
10 illustrates a hard disk drive 516 for reading from and writing to a non-removable,
11 non-volatile magnetic media (not shown), a magnetic disk drive 518 for reading
12 from and writing to a removable, non-volatile magnetic disk 520 (e.g., a "floppy
13 disk"), and an optical disk drive 522 for reading from and/or writing to a
14 removable, non-volatile optical disk 524 such as a CD-ROM, DVD-ROM, or other
15 optical media. The hard disk drive 516, magnetic disk drive 518, and optical disk
16 drive 522 are each connected to the system bus 508 by one or more data media
17 interfaces 526. Alternatively, the hard disk drive 516, magnetic disk drive 518,
18 and optical disk drive 522 may be connected to the system bus 508 by one or more
19 interfaces (not shown).

20 The disk drives and their associated computer-readable media provide non-
21 volatile storage of computer readable instructions, data structures, program
22 modules, and other data for computer 502. Although the example illustrates a hard
23 disk 516, a removable magnetic disk 520, and a removable optical disk 524, it is to
24 be appreciated that other types of computer readable media which may store data
25 that is accessible by a computer, such as magnetic cassettes or other magnetic

1 storage devices, flash memory cards, CD-ROM, digital versatile disks (DVD) or
2 other optical storage, random access memories (RAM), read only memories
3 (ROM), electrically erasable programmable read-only memory (EEPROM), and
4 the like, may also be utilized to implement the exemplary computing system and
5 environment.

6 Any number of program modules may be stored on the hard disk 516,
7 magnetic disk 520, optical disk 524, ROM 512, and/or RAM 510, including by
8 way of example, an operating system 526, one or more application programs 528,
9 other program modules 530, and program data 532.

10 A user may enter commands and information into computer 502 via input
11 devices such as a keyboard 534 and a pointing device 536 (e.g., a "mouse").
12 Other input devices 538 (not shown specifically) may include a microphone,
13 joystick, game pad, satellite dish, serial port, scanner, and/or the like. These and
14 other input devices are connected to the processing unit 504 via input/output
15 interfaces 540 that are coupled to the system bus 508, but may be connected by
16 other interface and bus structures, such as a parallel port, game port, or a universal
17 serial bus (USB). An example of an input/output interfaces 540 is an audio
18 system.

19 A monitor 542 or other type of display device may also be connected to the
20 system bus 508 via an interface, such as a video adapter 544. In addition to the
21 monitor 542, other output peripheral devices may include components such as
22 speakers (not shown) and a printer 546 which may be connected to computer 502
23 via the input/output interfaces 540.

24 Computer 502 may operate in a networked environment using logical
25 connections to one or more remote computers, such as a remote computing device

1 548. By way of example, the remote computing device 548 may be a personal
2 computer, portable computer, a server, a router, a network computer, a peer device
3 or other common network node, and the like. The remote computing device 548 is
4 illustrated as a portable computer that may include many or all of the elements and
5 features described herein relative to computer 502.

6 Logical connections between computer 502 and the remote computer 548
7 are depicted as a local area network (LAN) 550 and a general wide area network
8 (WAN) 552. Such networking environments are commonplace in offices,
9 enterprise-wide computer networks, intranets, and the Internet.

10 When implemented in a LAN networking environment, the computer 502 is
11 connected to a local network 550 via a network interface or adapter 554. When
12 implemented in a WAN networking environment, the computer 502 typically
13 includes a modem 556 or other means for establishing communications over the
14 wide network 552. The modem 556, which may be internal or external to
15 computer 502, may be connected to the system bus 508 via the input/output
16 interfaces 540 or other appropriate mechanisms. It is to be appreciated that the
17 illustrated network connections are exemplary and that other means of establishing
18 communication link(s) between the computers 502 and 548 may be employed.

19 In a networked environment, such as that illustrated with computing
20 environment 500, program modules depicted relative to the computer 502, or
21 portions thereof, may be stored in a remote memory storage device. By way of
22 example, remote application programs 558 reside on a memory device of remote
23 computer 548. For purposes of illustration, application programs and other
24 executable program components such as the operating system are illustrated herein
25 as discrete blocks, although it is recognized that such programs and components

1 reside at various times in different storage components of the computing device
2 502, and are executed by the data processor(s) of the computer.

3 Computer-Executable Instructions

4 An implementation of an exemplary CWD may be described in the general
5 context of computer-executable instructions, such as program modules, executed
6 by one or more computers or other devices. Generally, program modules include
7 routines, programs, objects, components, data structures, etc. that perform
8 particular tasks or implement particular abstract data types. Typically, the
9 functionality of the program modules may be combined or distributed as desired in
10 various embodiments.

12 Exemplary Operating Environment

13 Fig. 5 illustrates an example of a suitable operating environment 500 in
14 which an exemplary CWD may be implemented. Specifically, the exemplary
15 CWD(s) described herein may be implemented (wholly or in part) by any program
16 modules 528-530 and/or operating system 526 in Fig. 5 or a portion thereof.

17 The operating environment is only an example of a suitable operating
18 environment and is not intended to suggest any limitation as to the scope or use of
19 functionality of the exemplary CWD(s) described herein. Other well known
20 computing systems, environments, and/or configurations that are suitable for use
21 include, but are not limited to, personal computers (PCs), server computers, hand-
22 held or laptop devices, multiprocessor systems, microprocessor-based systems,
23 programmable consumer electronics, wireless phones and equipments, general-
24 and special-purpose appliances, application-specific integrated circuits (ASICs),
25

1 network PCs, minicomputers, mainframe computers, distributed computing
2 environments that include any of the above systems or devices, and the like.

3

4 Computer Readable Media

5 An implementation of an exemplary CWD may be stored on or transmitted
6 across some form of computer readable media. Computer readable media may be
7 any available media that may be accessed by a computer. By way of example, and
8 not limitation, computer readable media may comprise "computer storage media"
9 and "communications media."

10 "Computer storage media" include volatile and non-volatile, removable and
11 non-removable media implemented in any method or technology for storage of
12 information such as computer readable instructions, data structures, program
13 modules, or other data. Computer storage media includes, but is not limited to,
14 RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM,
15 digital versatile disks (DVD) or other optical storage, magnetic cassettes, magnetic
16 tape, magnetic disk storage or other magnetic storage devices, or any other
17 medium which may be used to store the desired information and which may be
18 accessed by a computer.

19 "Communication media" typically embodies computer readable
20 instructions, data structures, program modules, or other data in a modulated data
21 signal, such as carrier wave or other transport mechanism. Communication media
22 also includes any information delivery media.

23 The term "modulated data signal" means a signal that has one or more of its
24 characteristics set or changed in such a manner as to encode information in the
25 signal. By way of example, and not limitation, communication media includes

1 wired media such as a wired network or direct-wired connection, and wireless
2 media such as acoustic, RF, infrared, and other wireless media. Combinations of
3 any of the above are also included within the scope of computer readable media.

4

5 **Conclusion**

6 Although the invention has been described in language specific to structural
7 features and/or methodological steps, it is to be understood that the invention
8 defined in the appended claims is not necessarily limited to the specific features or
9 steps described. Rather, the specific features and steps are disclosed as preferred
10 forms of implementing the claimed invention.

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25